Bier Pongo

Kevin Fischer   
*OTH Amberg-Weiden*  
k.fischer2@oth-aw.de

Simon Kleber  
*OTH Amberg-Weiden*   
s.kleber1@oth-aw.de Markus Fleischmann  
*OTH Amberg-Weiden*   
m.fleischmann2@oth-aw.de

Fabian Klinger  
*OTH Amberg-Weiden*   
f.klinger1@oth-aw.de

Christian Renner  
*OTH Amberg-Weiden*   
c.renner@oth-aw.de Darren Fürst  
*OTH Amberg-Weiden*   
d.fuerst@oth-aw.de

Jakob Lindner  
*OTH Amberg-Weiden*   
j.lindner3@oth-aw.de

*Abstrakt*— Cloudfähige, skalierbare, multiuser- und multidevice-fähige Bier Pong Social Media Platform.

Schlüsselwörter—Bier Pong, Cloud, Data, Team

# Einleitung

In die Einleitung gehören eine Motivation und ein zusammenfassender Überblick Ihres Anwendungskonzeptes, ggf. mit fachlichen Alleinstellungsmerkmal („USPs"), sowie ggf. eine Benennung von technischen Schlüssel-Bausteinen. Die Einführungsabschnitt ist ebenfalls sehr gut geeignet Fachvokabular aus dem fachlichen Kontext der geplanten Anwendung einzuführen.

## Motivation

BLABLABLA

## Überblick

BLABLABLA

## Alleinstellungsmerkmal

BLABLABLA

## Technische Schlüsselbausteine

BLABLABLA

# Verwandte Arbeiten

## Abschnitt A

Im Abschnitt Verwandte Arbeiten können Sie bspw. andere Anwendungen oder Technologien aufführen, die Sie inspiriert haben. Auch Screenshots können der Veranschaulichung dienen (Bilder über beide Spalten hinweg sind in LaTeX und in Word möglich).

# Anforderungen

Einführung

## Erforderliche Anforderungen

### User Story 1

### Ich als \_\_\_ möchte \_\_\_ , um \_\_\_

**Akzeptanzkriterien:**

* Kriterium 1
* Kriterium 2

### User Story 2

### Ich als \_\_\_ möchte \_\_\_ , um \_\_\_

**Akzeptanzkriterien:**

* Kriterium 1
* Kriterium 2

## Optionale Anforderungen

### User Story 1

### Ich als \_\_\_ möchte \_\_\_ , um \_\_\_

**Akzeptanzkriterien:**

* Kriterium 1
* Kriterium 2

### User Story 2

### Ich als \_\_\_ möchte \_\_\_ , um \_\_\_

**Akzeptanzkriterien:**

* Kriterium 1
* Kriterium 2

# Technisches Grobkonzept

## Datenbank - AWS DynamoDB [1]

Eine AWS DynamoDB Datenbank soll zentraler Speicherort sein für alle Daten, wie z.B. laufende und abgeschlossne Spiele, die durch die Anwendung generiert wurden.

## REST API - AWS API Gateway [2]

Die in der AWS Cloud gehosteten Daten sollen über eine REST API im nötigen Format abrufbar sein. Diese REST API soll dabei durch einen AWS API Gateway realisiert werden, welche je nach Endpoint AWS Lambda Funktionen [3] aufruft.

## Back-End - AWS Lambda [3]

Alle nötigen Backend-Dienste sollen als AWS Lambda Funktionen realisiert werden. Durch diese werden Daten, wenn nötig transformiert, aus der DynamoDB gelesen und geschrieben.

## Benuterverwaltung - AWS Cognito [4]

AWS Cognito soll für eine Benutzerverwaltung genutzt werden, um nötige API Gateway Aufrufe zu authentifizieren und gleichzeitig bereits integrierte Funktionalität, wie Email Verifizierung, bereitzustellen.

## Web-App – React App [5]

Das Frontend soll React als Framework nutzen, um dem Benutzer eine Website bereitzustellen, welche die gesamte Funktionalität der Anwendung bietet. Ein AWS S3 Bucket [6] hostet diese als statische Website und macht sie so öffentlich zugänglich.

## Deployment - AWS CDK [7]

Um alle nötigen AWS Dienste zu konfigurieren und bereitzustellen soll das AWS Cloud Development Kit für Infrastructure as Code verwendet werden.

1. <https://aws.amazon.com/de/dynamodb/>
2. <https://aws.amazon.com/de/api-gateway/>
3. <https://aws.amazon.com/de/lambda/>
4. <https://aws.amazon.com/de/cognito/>
5. <https://reactjs.org/>
6. <https://aws.amazon.com/de/s3/>
7. <https://aws.amazon.com/de/cdk/>